

COMP 431 — INTERNET SERVICES & PROTOCOLS

Spring 2009

MW 9:30-10:45 AM, FB 007

Jasleen Kaur

jasleen@cs.unc.edu

<http://www.cs.unc.edu/~jasleen/Courses/Spring09>

January 12, 2009

READ THIS HANDOUT CAREFULLY. YOU ARE RESPONSIBLE FOR KNOWING THE GROUND-RULES FOR THIS COURSE!

Course Overview

The Internet has revolutionized the way large sections of the population communicate, research problems, shop, entertain themselves, and learn. But how does it work? How is it that a request for content can find its way from a web browser to an arbitrarily remote web server and the reply make it back to the requesting browser with no data being lost along the way? How do the hundreds of millions of web requests and responses that transit an Internet every second share the capacity of the network such that “traffic jams” do not occur? More importantly, how can one (or can one) control or improve the performance of their network connections?

This course provides a technical introduction to the architecture and operation of the Internet and the primary applications that are used on the Internet today. The goal is to understand how such basic services as e-mail, file transfer, and the World-Wide Web work and the principles involved in the design of such distributed services. We will study the myriad protocols and non-user-visible services involved in realizing these applications including consideration of the physical networks and switching elements that transmit 1s and 0s. Some of the specific topics we’ll study include:

- Application-level protocols HTTP, FTP, SMTP (e-mail), and the DNS (IP address resolution).
- Socket programming and client/server computing.
- Transport protocols TCP and UDP.
- Congestion control principles and algorithms.
- The Internet Protocol IP and the Internet routing architecture and algorithms.
- Physical-layer transmission technologies such as Asynchronous Transfer Mode (ATM), Ethernet, Wireless LANs, and modems, and interconnection technologies such as bridges, hubs, and switches.

Prerequisites

Prerequisites for this course are:

- COMP 411 (Computer Organization),
- COMP 410 (Data Structures),

- A working knowledge of the UNIX program development environment, and
- A working knowledge of the Java programming language.

These are *hard* prerequisites — *you will not survive in this course if you do not have these (or do not develop these within the first week of the course)!*

A “working knowledge of the UNIX program development environment” means a general familiarity with the UNIX operating system (*e.g.*, basic shell commands for file manipulation, I/O redirection, a file editor, compiler, and debugger) and basic operating system concepts such as processes, I/O, and buffering.

A “working knowledge of the Java programming language” means an ability to write a non-trivial program in Java on either UNIX or Windows. (For an example of a non-trivial program, see Homework 1.) Students not meeting these requirements must receive the explicit permission of the instructor to remain in this course.

Course Materials

The textbook for this course is:

Computer Networking: A Top-Down Approach Featuring the Internet, fourth edition, by James F. Kurose and Keith W. Ross, Addison Wesley, 2008.

Time permitting, we will attempt to cover all of chapters 1-5.

This course has a significant “laboratory” component. You will be asked write several programs that illustrate the main concepts of TCP socket programming and client/server interaction. These programs will be written in Java on UNIX. You will also have assignments in which you use network monitoring and debugging tools to examine various aspects of Internet protocols.

If you do not already own a Java text, two good Java references are:

- *The Java Programming Language*, third edition, by Ken Arnold, James Gosling, and David Holmes, Addison Wesley, 2000,
- *Java in a Nutshell*, third edition, by David Flanagan, O’Reilly & Associates, 1999.

In addition to these texts, lecture notes will be handed out at the start each class. If you miss class, copies of lecture notes, as well as copies of all class handouts, homeworks, assignments, *etc.*, can be found on the course web page at <http://www.cs.unc.edu/~jasleen/Courses/Spring09>.

Grading

Your grade in this class will be based on:¹

- Written homework, laboratory and programming assignments (40%),
- Midterm examination(s) (25%),
- Final examination (35%), and
- Class participation.

Students’ participation in class may be used to either adjust the final course grade up by a “+” (*e.g.*, from a B to a B+) or down by a “-” (*e.g.*, from a C to a C-).

¹ The percentages listed are only approximate and are subject to change (by no more than 10%).

Homework Assignments

Homework will be assigned approximately every one to two weeks and take one to two weeks to complete. Hence you will always have an outstanding homework assignment in this course. Assignments will be either laboratory (programming) exercises or written word problems. The written homework assignments will have questions and problems similar to those given on the midterm and final exams. All written assignments will be collected at the *start of class* (within the first 5 minutes) on the day on which they are due. All programming assignments will be submitted electronically and must be submitted *before* the start of class on the day on which they are due. Written homework will be collected by the instructor; programming assignments will be submitted by copying files to a special directory on a departmental UNIX machine. (The exact procedure will be described on the first homework handout.)

All homework submitted after its deadline is considered late. Assignments that are submitted within 24 hours after the original deadline are considered to be “one day late,” within 48 hours, “two days late,” *etc.* A late homework assignment will be accepted without penalty only if the total “lateness” of all homework assignments received to date (including the current assignment) does not exceed 5 days. If this condition is not met then the student will receive a *failing grade* on the assignment. Once a student’s 5 “late days” have been used up *no extensions for subsequent late homeworks will be granted*. The student will receive a failing grade for all subsequent late homework assignment (the late homework assignments will be graded in order to provide feedback to the student, however, a failing grade will be recorded for the assignment).

Late programs should be submitted as usual by the student should send email to the TA informing her that the program is available for grading. (If email is not sent to the TA the student runs the risk that the program may not be graded.) Late written assignments must be *hand delivered* to either the instructor, the TA, or Ms. Dorothy Turner (in FB 114). In the case of late written assignments, the lateness of the assignment will be assessed based on when the assignment is *received* and *not* on when it is submitted. Thus, for example, if you complete a late assignment on a Saturday, it will be considered to have been completed when it is received on the following Monday.

Students who submit all homework assignments on time (no late days used) will receive a 10% homework grade bonus at the end of the semester.

Exams

There will be one midterm exam. It is tentatively scheduled to be given on March 4. The final exam for this course is scheduled for Monday, May 4, 2009 at 8:00 AM. The final exam will be comprehensive and covers the entire course.

Office Hours

I will hold office hours from 12:15 – 1:15 on Mondays and Wednesdays in FB 136. The TAs office hours will be set later this week.

Course Conduct and the Honor Code

Students are *encouraged* to work together on written homework assignments. *Acceptable collaboration* on written homework assignments includes:

- Discussing the assigned problems to understand their meaning,
- Discussing possible approaches to assigned problems,
- Obtaining help with issues in using operating system functions, Java syntax or semantics, or programs used for laboratory assignments.

In all cases you must *explicitly* acknowledge any and all substantive help received from other individuals during the course of the preparation of your homework solution. That is, if you collaborate with other individuals then you *must* include an explicit acknowledgment in your homework solution of the persons from whom you received aid. You should include the acknowledgement with your Honor Code pledge. Acknowledging others, if done properly, will not adversely affect your grade.

Unacceptable collaboration on written homework includes:

- Copying (verbatim use) of physical papers or computer files,²
- Submission of solutions that are jointly authored, or authored either wholly or in part by other individuals,
- Having someone else debug your program.

The general rule to be followed is that the strategy and approach of solutions may be developed jointly but *all* actual solutions (*i.e.*, the final solution) must be *constructed* and *written up* individually. Work done jointly should not be done in sufficient detail as to make it a final solution. For example, solutions may *sketched* out jointly, however each student must construct the final form of their solution individually and write-up or program their own solution.

For programming assignments *absolutely no code may be shared between students*. If you copy or transcribe one or more lines of code from any other student (including students or persons not in this course) you are in violation of the Honor Code.

Any form of unacceptable collaboration will be considered a violation of the UNC Honor Code and will be reported to the Student Attorney General.

Should questions arise the course of working on a problem please feel free to immediately contact the instructor either by telephone, electronic mail, or by an office visit. In principle, if you work with others in good faith and are honest and generous with your attributions of credit you will have no problems.

² This includes computer files that are copied and then edited and/or reformatted.

Course Outline

The course will closely follow the Kurose and Ross text. The goal is to complete at least Chapters 1-4 and as much of Chapter 5 as time permits. The following outline is taken from the text.

Chapter 1: Computer Networks and the Internet

- 1.1: What Is the Internet?
- 1.2: The Network Edge
- 1.3: The Network Core
- 1.4: Access Networks and Physical Media
- 1.5: ISPs and Internet Backbones
- 1.6: Delay and Loss in Packet-Switched Networks
- 1.7: Protocol Layers and Their Service Models
- 1.8: History of Computer Networking and the Internet

Chapter 2: Application Layer

- 2.1: Principles of Network Application
- 2.2: The Web and HTTP
- 2.3: File Transfer: FTP
- 2.4: Electronic Mail in the Internet
- 2.5: DNS--The Internet's Directory Service
- 2.6: P2P File Sharing
- 2.7: Socket Programming with TCP
- 2.8: Socket Programming with UDP
- 2.9: Building a Simple Web Server

Chapter 3: Transport Layer

- 3.1: Introduction and Transport-Layer Services
- 3.2: Multiplexing and Demultiplexing
- 3.3: Connectionless Transport: UDP
- 3.4: Principles of Reliable Data Transfer
- 3.5: Connection-Oriented Transport: TCP
- 3.6: Principles of Congestion Control
- 3.7: TCP Congestion Control

Chapter 4: Network Layer and Routing

- 4.1: Introduction
- 4.2: Virtual Circuit and Datagram Networks
- 4.3: What's Inside a Router?

4.4: The Internet Protocol (IP): Forwarding and Addressing in the Internet

4.5: Routing Algorithms

4.6: Routing in the Internet

4.7: Broadcast and Multicast Routing

Chapter 5: Link Layer and Local Area Networks

5.1: Link Layer: Introduction and Services

5.2: Error-Detection and -Correction Techniques

5.3: Multiple Access Protocols

5.4: Link-Layer Addressing

5.5: Ethernet

5.6: Interconnections: Hubs and Switches

5.7: PPP: The Point-to-Point Protocol

5.8: Link Virtualization: A Network as a Link Layer